

4. előadás

A táblázat

A táblázatról általában, soros, önátrendező, rendezett és kulcstranszformációs táblázat

Adatszerkezetek és algoritmusok előadás
2018. február 27.

A táblázat

Kósa Márk
Pánovics János
Szathmáry László



Asszociatív
adatszerkezetek

A táblázatról általában

Soros táblázat

Önátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

Szinonimák

Kósa Márk, Pánovics János és Szathmáry László
Debreceni Egyetem
Informatikai Kar



Adatelemei tartalmuk alapján, valamilyen kulcs segítségével címezhetőek (pl.: tömb, mátrix, táblázat).

Legegyszerűbb eset: kulcs = index (pl. tömb).

Megkülönböztetünk:

- 1 közvetlen (direkt) szervezésű struktúrák
- 2 véletlen (indirekt) szervezésű struktúrák

Feladat: a kulcs alapján feleltessünk meg valamilyen memóriaterületet az adatalemnek.

Közvetlen szervezésű struktúrák esetén a kulcs és az adatalem között egyértelmű megfeleltetés állítható fel.

Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Ónátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

Szinonimák



A táblázat **dinamikus**, **homogén** és **asszociatív** adatszerkezet.

| KULCS | ÉRTÉK |
|-------|-------|
| | |
| | |
| | |
| | |
| | |
| | |

← a táblázat
egy eleme

Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Önátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

Szinonimák

Elemei **összetettek**: **kulcs** és **érték** komponensből állnak.

A kulcsnak a táblázatban előforduló értékeit **kulcsértékeknek** nevezzük.

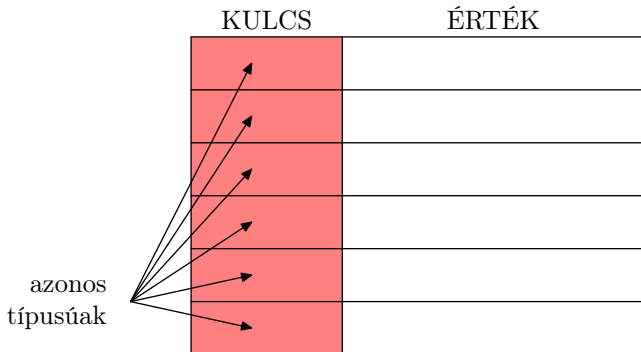
| | KULCS | ÉRTÉK |
|----------------------------------|-------|-------|
| a kulcs értékei (különbözőek) | | |
| | | |
| | | |
| | | |
| | | |
| | | |

A táblázat elemeit a kulcsuk értékei alapján különböztetjük meg egymástól. A táblázatban **nem szerepelhet** két **azonos** kulcsérték. (Ezért nevezhetjük őket kulcsértékeknek.)



A táblázat

A **táblázat** az egydimenziós tömb **általánosításának** tekinthető: az adatelemeket a kulcsuk értéke alapján azonosítjuk, ezeknek az értékeknek azonban nem feltétlenül kell egész számoknak lenniük, lehet más típusuk is.

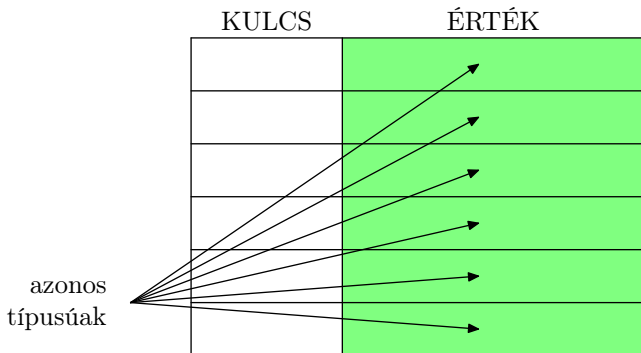


A **KULCS** és **ÉRTÉK** komponensek típusának sem feltétlenül kell megegyeznie egymással. Az adatelemekben a kulcsértékek típusát a KULCS típusa, az értékek típusát az ÉRTÉK típusa határozza meg.



A táblázat

A **táblázat** az egydimenziós tömb **általánosításának** tekinthető: az adatelemeket a kulcsuk értéke alapján azonosítjuk, ezeknek az értékeknek azonban nem feltétlenül kell egész számoknak lenniük, lehet más típusuk is.



A **KULCS** és **ÉRTÉK** komponensek típusának sem feltétlenül kell megegyeznie egymással. Az adatelemekben a kulcsértékek típusát a KULCS típusa, az értékek típusát az ÉRTÉK típusa határozza meg.



Soros táblázat

Az adatelemek táblázatbeli sorrendje megegyezik az elemek táblázatba kerülésének időbeli (érkezési) sorrendjével.

Soros táblázattal végezhető műveletek

- **Létrehozás:** a KULCS és az ÉRTÉK típusának meghatározása, és az elemek elhelyezése érkezési sorrendben.
- **Bővítés:** a táblázat végén.
- **Törlés:** mind folytonos, mind szétszórt reprezentáció mellett megvalósítható a fizikai törlés.
- **Csere:** az érték rész bármikor, kulcsérték csak akkor cserélhető, ha az új érték még nem szerepel.
- **Rendezés:** nem értelmezett.
- **Keresés:** teljes keresés a kulcs értéke alapján.
- **Elérés:** soros.
- **Bejárás:** általában az elejétől a végéig.
- A **feldolgozás** alapja a kulcs és a teljes keresés.



Folytonosan és szétszórtan (pl. egyirányban láncolt listával) egyaránt ábrázolható.

Előbbi esetben a **fizikai törlést** úgy valósíthatjuk meg, hogy a törlendő adatelemet a táblázat utolsó elemével felülírjuk, és csökkentjük a táblázat aktuális elemszámát.

Utóbbi esetben a törlés – szokás szerint – mutatóértékek cseréjével oldható meg.

Mikor érdemes használni?

Ha az elemek feldolgozási gyakorisága nagyjából azonos, és nem lényeges az elemek feldolgozásának a sorrendje.



Mikor érdemes használni?

Ha az elemek feldolgozási gyakorisága eltérő.

Önátrendező táblázattal végezhető műveletek

A műveletek legtöbbje megegyezik a soros táblázatével.
Kivételt képez a **feldolgozás**.

- Egy elem **feldolgozása** után a feldolgozott elemet a táblázat elejére helyezzük, az addigi első elem elé. Ennek következtében mindig a legutoljára feldolgozott elem lesz a táblázat elején.

Nagyszámú adatelem feldolgozása után az elemek sorrendje a táblázatban jól fogja közelíteni a feldolgozási gyakoriságot: a táblázat elején lesznek a gyakrabban feldolgozott elemek.

Az önátrendező táblázat legjobban szétszórtnan, egyirányban láncolt listával reprezentálható. Ekkor a feldolgozás művelete mindössze három mutató értékének a cseréjét jelenti.





Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Önatrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing
Kulcstranszformációs
módszerek
Szinonimák

Rendezett táblázat

Elemei a kulcsértékek alapján rendezettek, az adatelemek sorrendjét a kulcsértékek (általában) növekvő sorrendje definiálja.

Rendezett táblázattal végezhető műveletek

A műveletek legtöbbje megegyezik a soros táblázatéval. Kivételt képeznek a következő műveletek:

- **Létrehozás** és **bővítés**: az érkező elemeket rendezetten helyezzük el a táblázatban, rendezett sorozatba történő beszúrással.
- **Csere**: az adatelem érték része bármikor cserélhető.
- **Keresés**: lineáris vagy bináris (reprezentációfüggő).

Folytonos reprezentációnál gyorsabb a keresés, de nehézkes a bővítés és a törlés. Szétszórt reprezentációnál ennek az ellenkezője igaz.

Mikor érdemes használni?

Ha az elemek feldolgozási gyakorisága nagyjából azonos, és fontos a feldolgozásuk sorrendje és gyorsasága.

Kulcstranszformációs táblázat

Véletlen szervezésű struktúrák.

Cél

A közvetlen elérés biztosítása a kulcsértékek alapján.

Hash függvény

A kulcstranszformációs táblázatban egy K értékű kulccsal rendelkező elem helyét (címét) egy h függvény $h(K)$ értéke határozza meg. Ezt a h függvényt hívjuk **hasító** vagy **hash** függvénynek.

Hashing

Azt az eljárást, melynek során egy adatelem K értékű kulcsához meghatározzuk a $h(K)$ értéket (az adatelem táblázatbeli helyét), **hasításnak**, **hashingnek**, **randomizálásnak** vagy **kulcstranszformációnak** nevezzük.

Ennek a táblázatfajtának az ábrázolása **folytonos**.





Kölcsönösen egyértelmű hash függvény használható, ha

- a gyakorlatban előforduló kulcsértékek száma közel azonos az elvileg lehetséges kulcsértékek számával;
- a gyakorlatban előforduló kulcsértékek egyenletesen oszlanak el az elvileg lehetséges kulcsértékek között.

Ha a gyakorlatban előforduló kulcsértékek száma és az elvi lehetőségek száma között nagy az eltérés, és a gyakorlatban előforduló kulcsértékeknek nem egyenletes az eloszlása, akkor csak **egyértelmű** hash függvények használatára van lehetőség.

Egy egyértelmű hash függvénytől a következőket várjuk el:

- a gyakorlatban előforduló kulcsokat képezze le a rendelkezésre álló címtartományba;
- a rendelkezésre álló címtartományon belül tegye *egyenletessé* az elemek eloszlását.

Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Önatrendező táblázat

Rendezett táblázat

Kulcstranzformációs
táblázat

Hash függvény, hashing

Kulcstranzformációs
módszerek
Színanimák

Kulcstranszformációs módszer

A kulcstranszformációs módszer egy algoritmus, amely azt írja le, hogy a hash függvény hogyan képezi le a kulcsértéket a tárbeli címre.

A gyakorlatban a kulcsok típusa alapján megkülönböztetünk

- **szöveges** és
- **numerikus** kulcsokat.

Szöveges típusú kulcsok esetén a szöveget alkotó karakterek belső kódjainak valamilyen numerikus függvényét tekintjük, amellyel a kulcstranszformációt a numerikus típusú kulcsok esetére vezethetjük vissza.

Numerikus típusú kulcsok esetén – többek között – az alábbi módszerek használhatók:

- prímszámmal való osztás
- szorzás
- helyiérték-kiválasztás
- bázistranszformáció



[Asszociatív adatszerkezetek](#)

[A táblazatról általában](#)

[Soros táblázat](#)

[Őnatrendező táblázat](#)

[Rendezett táblázat](#)

[Kulcstranszformációs táblázat](#)

[Hash függvény, hashing](#)

[Kulcstranszformációs módszerek](#)

[Szinonimák](#)

Univerzális hash függvény nem létezik.

Szinonimák

A csak egyértelmű hash függvényeknél előfordulhat, hogy a függvény a **különböző** kulcsértékekkel rendelkező adatelemekhez **ugyanazt** a tárcímet rendeli. Az ilyen adatelemeket **szinonimáknak**, magát a jelenséget pedig **túlcsordulásnak** nevezzük.

(Egyéb elnevezések: ütközés, ütközéskezelés.)

A szinonimák felbukkanását kezelni kell, mert nem helyezhetünk el két vagy több adatelemet ugyanazon a tárhelyen.

Szinonimakezelő módszerek:

- nyílt címzés módszere
- nyílt címzés (belső) láncolással
- független túlcsordulási lista alkalmazása

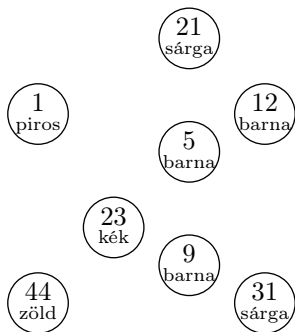


Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k)$$

| | |
|--|--|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

$$h(k) = k \bmod 11 + 1$$



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Őnátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

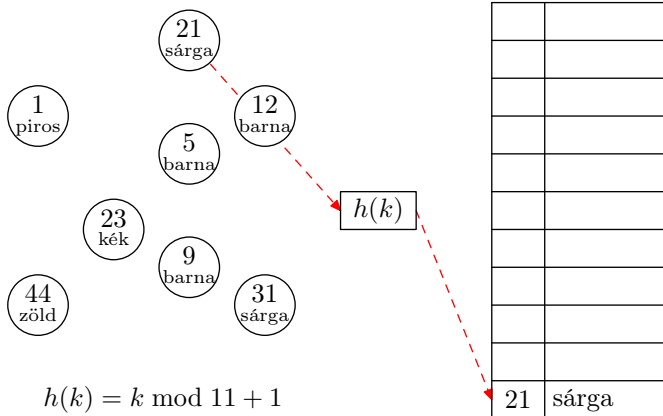
Szinonimák

Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:

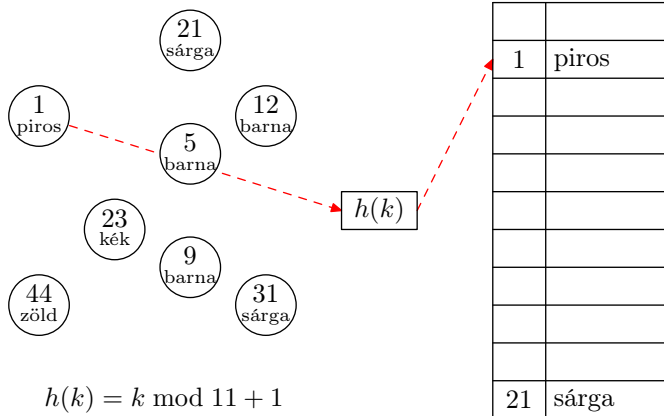


Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Őnátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

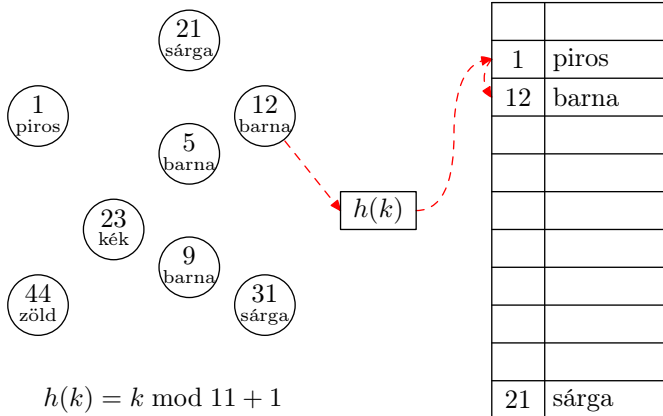
Szinonimák

Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Őnátrendező táblázat

Rendezett táblázat

Kulcsstranszformációs
táblázat

Hash függvény, hashing

Kulcsstranszformációs
módszerek

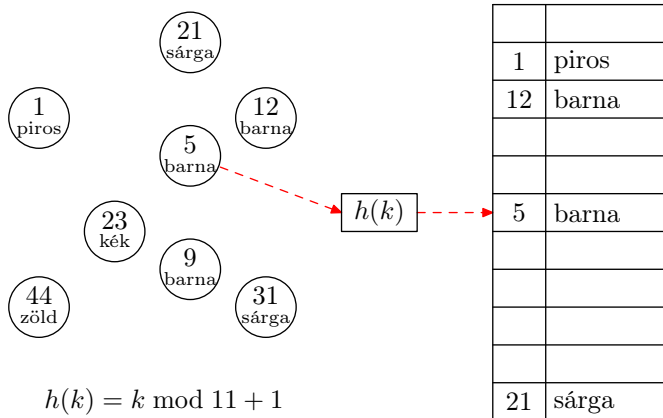
Szinonimák

Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:

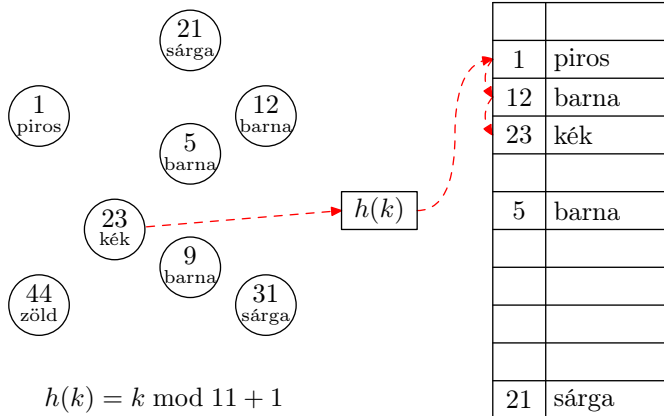


Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Őnátrendező táblázat

Rendezett táblázat

Kulcsstranszformációs
táblázat

Hash függvény, hashing

Kulcsstranszformációs
módszerek

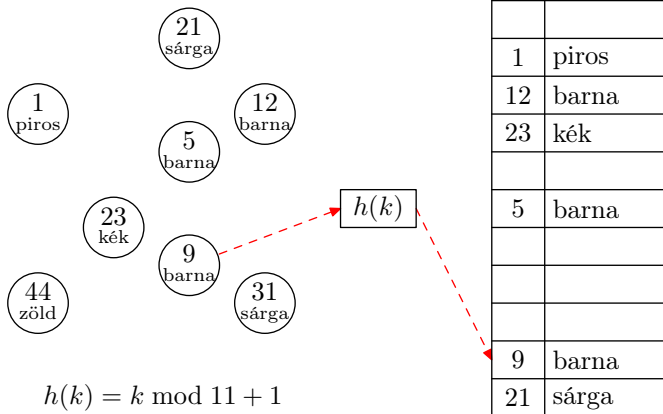
Szinonimák

Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:

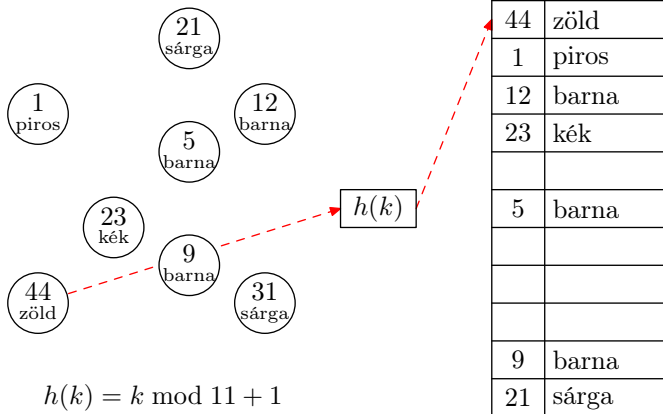


Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:

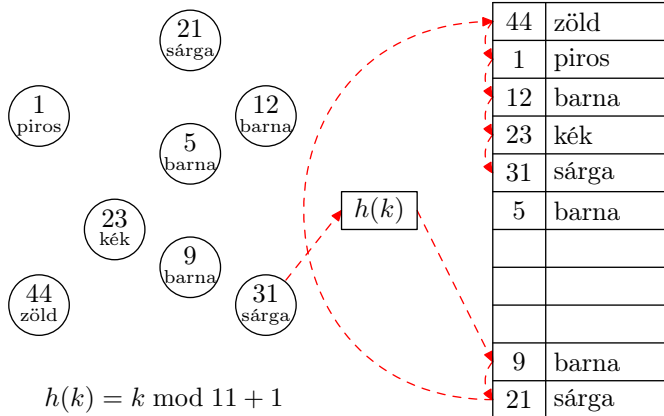


Nyílt címzés módszere

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:

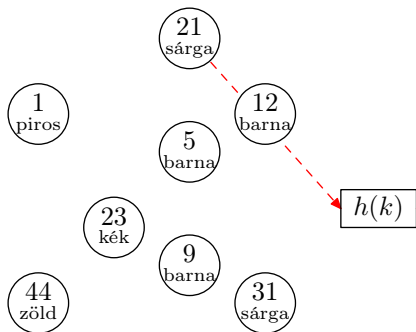


Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | 0 |

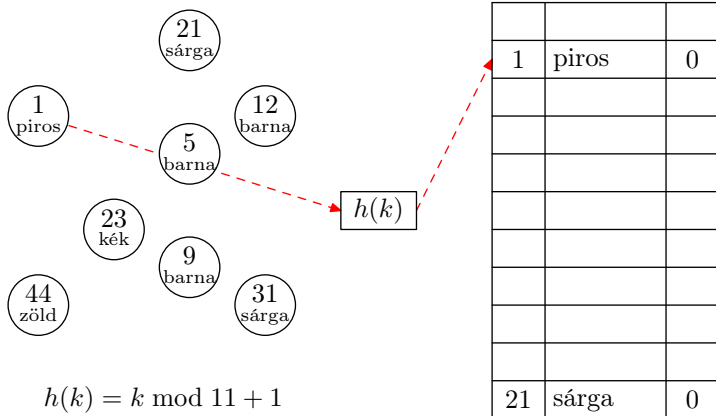


Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Önátrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

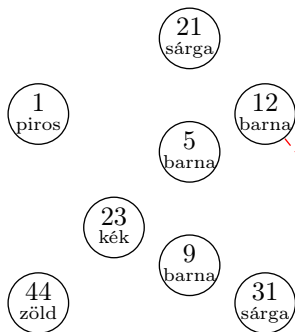
Szinonimák

Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| | | |
| 1 | piros | 3 |
| 12 | barna | 0 |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | 0 |

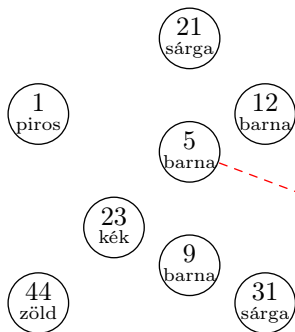


Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| | | |
| 1 | piros | 3 |
| 12 | barna | 0 |
| | | |
| | | |
| 5 | barna | 0 |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | 0 |

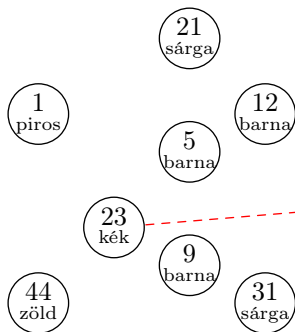


Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| | | |
| 1 | piros | 3 |
| 12 | barna | 4 |
| 23 | kék | 0 |
| | | |
| 5 | barna | 0 |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | 0 |



Asszociatív
adatszerkezetek

A táblazatról általában

Soros táblázat

Őnatrendező táblázat

Rendezett táblázat

Kulcstranszformációs
táblázat

Hash függvény, hashing

Kulcstranszformációs
módszerek

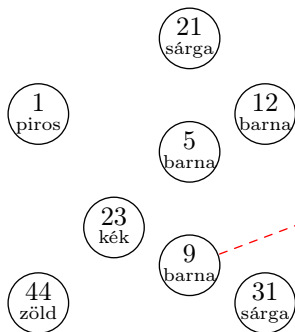
Szinonimák

Ült címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| | | |
| 1 | piros | 3 |
| 12 | barna | 4 |
| 23 | kék | 0 |
| | | |
| 5 | barna | 0 |
| | | |
| | | |
| | | |
| | | |
| 9 | barna | 0 |
| 21 | sárga | 0 |

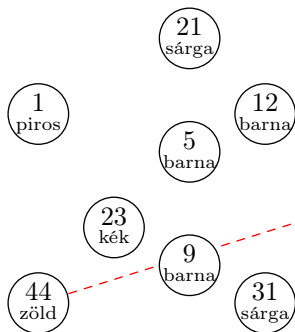


Nyílt címzés (belső) láncolással

Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|---|
| 44 | zöld | 0 |
| 1 | piros | 3 |
| 12 | barna | 4 |
| 23 | kék | 0 |
| | | |
| 5 | barna | 0 |
| | | |
| | | |
| | | |
| 9 | barna | 0 |
| 21 | sárga | 0 |

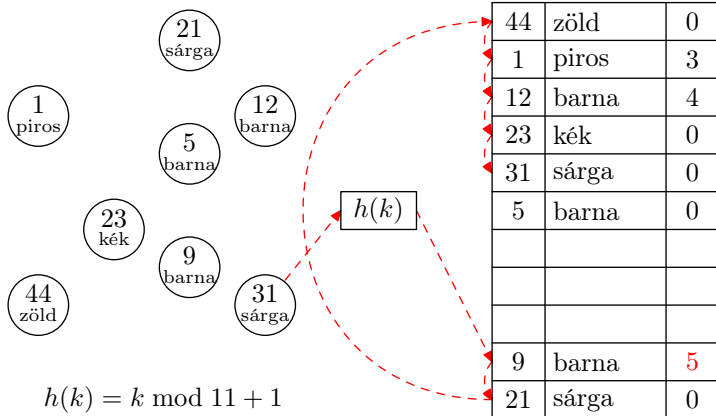


Ílt címzés (belső) láncolással

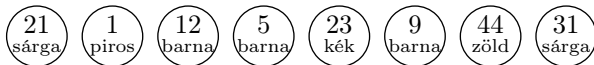
Egy tárhely státusza lehet:

- szabad (üres vagy logikailag törölt)
- foglalt

A táblázat bővítése:



Független túlcsoordulási lista

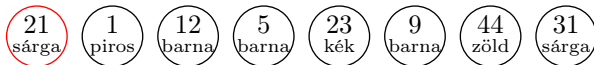


$$h(k) = k \bmod 11 + 1$$

| | | |
|--|--|--|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |



Független túlcsoordulási lista

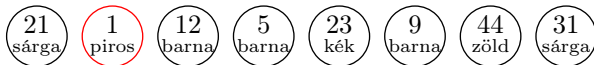


$$h(k) = k \bmod 11 + 1$$

| | | |
|----|-------|-----|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | NIL |



Független túlcsoordulási lista

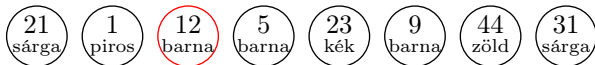


$$h(k) = k \bmod 11 + 1$$

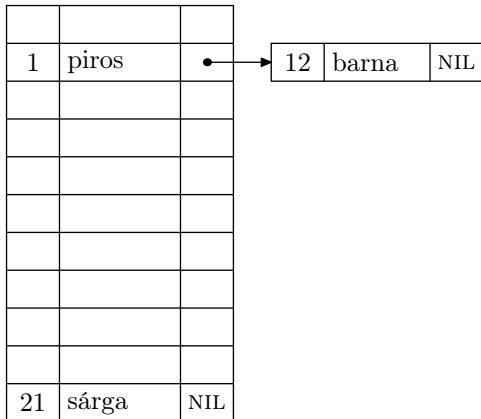
| | | |
|----|-------|-----|
| | | |
| 1 | piros | NIL |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| 21 | sárga | NIL |



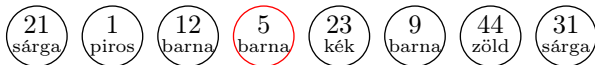
Független túlcsoordulási lista



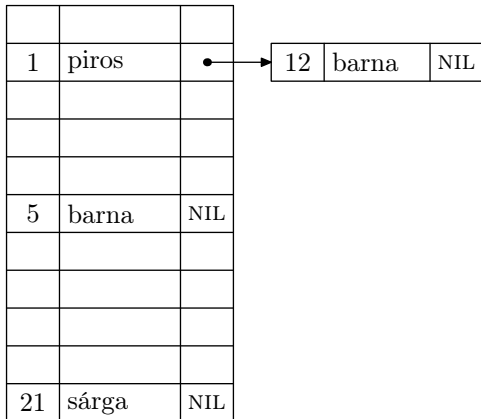
$$h(k) = k \bmod 11 + 1$$



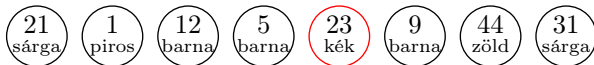
Független túlcsoordulási lista



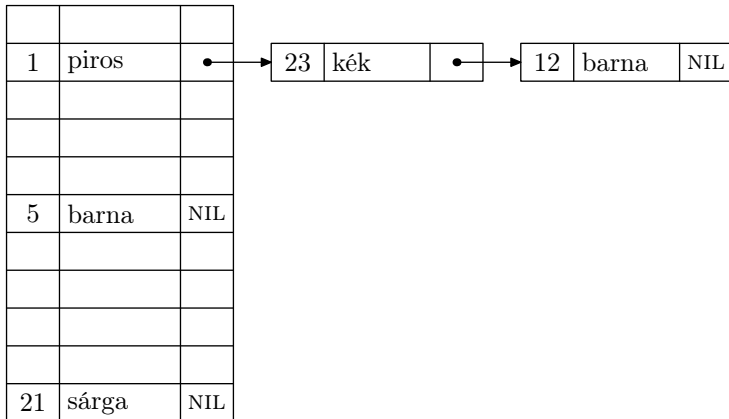
$$h(k) = k \bmod 11 + 1$$



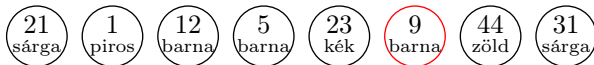
Független túlcsoordulási lista



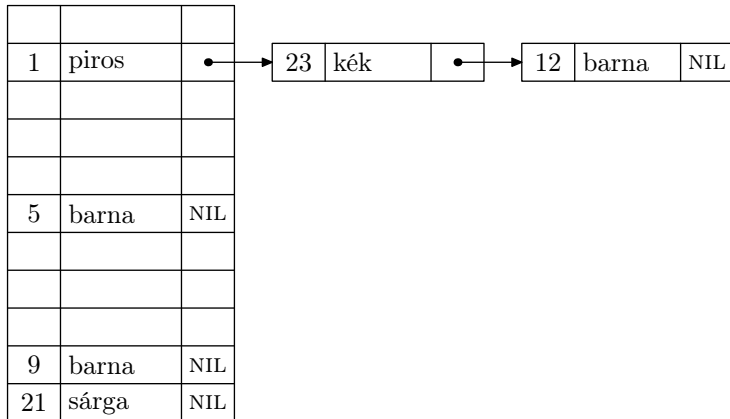
$$h(k) = k \bmod 11 + 1$$



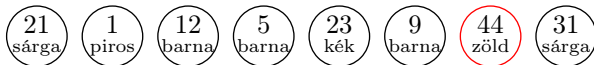
Független túlcsoordulási lista



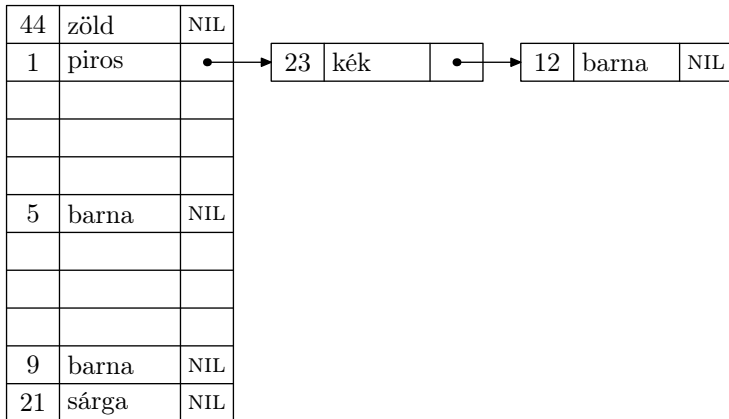
$$h(k) = k \bmod 11 + 1$$



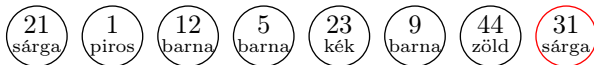
Független túlcsoordulási lista



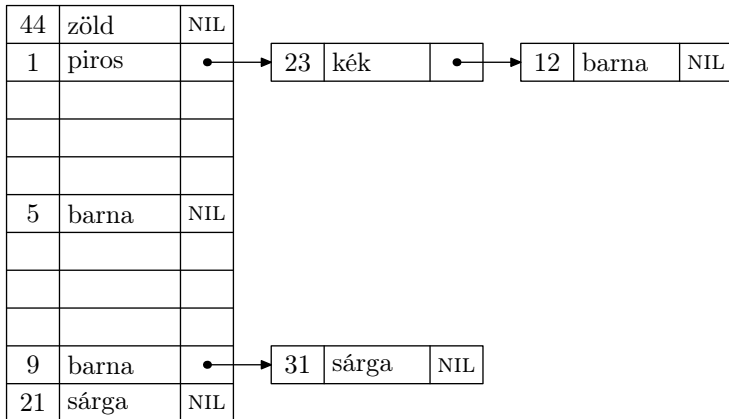
$$h(k) = k \bmod 11 + 1$$



Független túlcsordulási lista



$$h(k) = k \bmod 11 + 1$$



Kulcstranszformációs táblázattal végezhető műveletek

- **Létrehozáskor** a gyakorlatban előforduló kulcsértékek darabszámát megbecsüljük, kiválasztunk egy hash függvényt és (ha szükséges) egy szinonimakezelési módszert, majd lefoglaljuk a tárhelyeket a becslésnek megfelelően, és mindegyiket üresre állítjuk.
- **Bővítés:** az adatalem kulcsértéke alapján, a hash függvény segítségével.
- **Törlés:** szinte kizárólag logikai; fizikai csak a túlcsoportulási listából.
- **Csere:** kulcs alapján a hozzá tartozó értéket lehet cserélni.
- **Rendezés:** nincs.
- **Elérés:** kvázi közvetlen, a hash függvény a közvetlen elérést szolgálja, a szinonimákat keresni kell.
- **Keresés:** általában nincs, csak szinonimákat kereshetünk, az pedig a szinonimakezelési módszertől függ.
- **Bejárás:** nincs, mivel csak közvetlen elérés van.
- A **feldolgozás** alapja a hash függvény, illetve a közvetlen elérés.

